

Page 1, line 9, delete "The Prior Art" and substitute: --DESCRIPTION OF RELATED ART--;

Page 2, at line 29 delete "Presentation of the Drawings, and substitute the following heading at the left hand margin: --BRIEF DESCRIPTION OF THE DRAWINGS--;

Page 3, at line 23 delete "Detailed Description of Examples Illustrating the Invention" and substitute the following heading at the left hand margin: --DESCRIPTION OF THE PREFERRED EMBODIMENT(S)--;

Page 34, after line 8, insert the following new paragraph:

--While this invention has been described in conjunction with specific embodiments thereof, it is evident that many alternatives, modifications and variations will be apparent to those skilled in the art. Accordingly, the preferred embodiments of the invention as set forth herein, are intended to be illustrative, not limiting. Various changes may be made without departing from the true spirit and full scope of the invention as set forth herein and defined in the claims.—

IN THE CLAIMS:

Please cancel Claims 1 - 6 in their entirety and without prejudice and substitute the following new claims:

- 1 --7. A method for controlling a function (f) executable by various software
- 2 products (4a, 4b, 6a, 6b) by means of commands (Pa-Pd) specific to the respective
- 3 software products and each capable of having at least one option, the software
- 4 products being installed in at least one machine (2a, 2b) of a computer system (10),
- 5 comprising defining in an abstract class an abstract method for the function (f), the
- 6 abstract method including parameters corresponding to a union, in the logical sense,

7 of all options (Table C) of a specific command, defining a common command (P0)  
8 that includes arbitrary symbols corresponding to parameters of the abstract method,  
9 creating at least one driver (52) for implementing the abstract method in a machine,  
10 and executing by the driver one of the specific commands with options equivalent to  
11 the options of the common command.

1 8. A method according to claim 7, wherein equivalence between options of  
2 the specific command and options of the common command comprises creating a  
3 configuration file (ConfigPrint) defining types and default values of the options of  
4 each specific command that can be executed by the driver, and determining  
5 parameters of one of said specific commands by consulting a configuration file by  
6 means of the common command.

1 9. A method according to claim 7, wherein a driver (52a) corresponds to a  
2 machine (2a) of the computer system.

1 10. A method according to claim 8, wherein a driver (52a) corresponds to a  
2 machine (2a) of the computer system.

1 11. A method according to claim 7, wherein the abstract class is the most  
2 abstract class that can be defined, such as an interface in the Java® language.

1 12. A method according to claim 8, wherein the abstract class is the most  
2 abstract class that can be defined, such as an interface in the Java® language.

1 13. A method according to claim 9, wherein the abstract class is the most  
2 abstract class that can be defined, such as an interface in the Java® language.

1 14. A method according to claim 10, wherein the abstract class is the most  
2 abstract class that can be defined, such as an interface in the Java® language.

1 15. A method according to claim 7, wherein the abstract class contains at least  
2 some of the methods relating to functions of a functionality (F) common to the  
3 software products.

1 16. A method according to claim 8, wherein the abstract class contains all or  
2 some of the methods relating to functions of a functionality (F) common to the  
3 software products.

1 17. A method according to claim 9, wherein the abstract class contains all or  
2 some of the methods relating to functions of a functionality (F) common to the  
3 software products.

1 18. A method according to claim 10, wherein the abstract class contains all or  
2 some of the methods relating to functions of a functionality (F) common to the  
3 software products.

1 19. A method according to claim 11, wherein the abstract class contains all or  
2 some of the methods relating to functions of a functionality (F) common to the  
3 software products.

1 20. A computer system (10) comprising at least one machine having various  
2 software products (4a, 4b, 6a, 6b) having in common at least one function (f)  
3 executable by means of commands (Pa-Pd) specific to the respective software  
4 products and each capable of having at least one option, and adapted to implement a  
5 method for controlling a function (f) executable by various software products (4a, 4b,  
6 6a, 6b) by means of commands (Pa-Pd) specific to the respective software products  
7 and each capable of having at least one option, the software products being installed in  
8 at least one machine (2a, 2b) of a computer system (10), means for defining in an  
9 abstract class an abstract method for the function (f), the abstract method including  
10 parameters corresponding to a union, in the logical sense, of all options (Table C) of a  
11 specific command, means for defining a common command (P0) that includes  
12 arbitrary symbols corresponding to parameters of the abstract method, means for  
13 creating at least one driver (52) for implementing the abstract method in a machine,  
14 and means for executing by the driver one of the specific commands with options  
15 equivalent to the options of the common command.

1 21. A computer system, according to claim 26, further comprising means for  
2 creating a configuration file (ConfigPrint) defining the types and the default values of  
3 the options of each specific command that can be executed by the driver, and means  
4 determining the parameters of one of said specific commands by consulting a

5 configuration file by means of the common command so as to provide equivalence  
 6 between the options of the specific command and the options of the common  
 7 command.

1 22. A computer system according to claim 20 wherein a machine (2a) of the  
 2 computer system includes a drive (52a).

1 23. A computer system according to claim 20 wherein the abstract class is the  
 2 most abstract class that can be defined.

1 24. A computer system according to claim 20 wherein the abstract class  
 2 contains all or some of the methods relating to functions of a same functionality (F)  
 3 common to the software products.

1 25. A computer system as set forth in claim 23 wherein the abstract class is an  
 2 interface in the Java® language.

### IN THE ABSTRACT:

Please amend the Abstract at page 37 by deleting the last line "Fig. 1 to be published".

### REMARKS

This Supplemental Preliminary Amendment is filed to insert headings to conform the application to U.S. practice, to correct informalities in the specification,